Probabilistic Causal Effect Estimation With Global Neural Network Forecasting Models

Article *in* IEEE Transactions on Neural Networks and Learning Systems · July 2022

DOI: 10.1109/TNNLS.2022.3190984

7 authors, including:

Priscila Grecov
Monash University (Australia)
5 PUBLICATIONS **9** CITATIONS
SEE PROFILE

Samuel Campbell
Monash University (Australia)
8 PUBLICATIONS **54** CITATIONS
SEE PROFILE

Debbie Scott
Deakin University
112 PUBLICATIONS **1,240** CITATIONS
SEE PROFILE

Dan I Lubman
Monash University (Australia)
658 PUBLICATIONS **21,127** CITATIONS
SEE PROFILE

Probabilistic Causal Effect Estimation With Global Neural Network Forecasting Models

Priscila Grecov, Ankitha Nandipura Prasanna, Klaus Ackermann, Sam Campbell, Debbie Scott, Dan I. Lubman, and Christoph Bergmeir

***Abstract*— We introduce a novel method to estimate the causal effects of an intervention over multiple treated units by combining the techniques of probabilistic forecasting with *global* forecasting methods using deep learning (DL) models. Considering the counterfactual and synthetic approach for policy evaluation, we recast the causal effect estimation problem as a counterfactual prediction outcome of the treated units in the absence of the treatment. Nevertheless, in contrast to estimating only the counterfactual time series outcome, our work differs from conventional methods by proposing to estimate the counterfactual time series probability distribution based on the past preintervention set of treated and untreated time series. We rely on time series properties and forecasting methods, with shared parameters, applied to stacked univariate time series for causal identification. This article presents *DeepProbCP*, a framework for producing accurate quantile probabilistic forecasts for the counterfactual outcome, based on training a *global* autoregressive recurrent neural network model with conditional quantile functions on a large set of related time series. The output of the proposed method is the counterfactual outcome as the spline-based representation of the counterfactual distribution. We demonstrate how this probabilistic methodology added to the *global* DL technique to forecast the counterfactual trend and distribution outcomes overcomes many challenges faced by the baseline approaches to the policy evaluation problem. Oftentimes, some target interventions affect only the tails or the variance of the treated units' distribution rather than the mean or median, which is usual for skewed or heavy-tailed distributions. Under this scenario, the classical causal effect models based on counterfactual predictions are not capable of accurately capturing or even seeing policy effects. By means of empirical evaluations of synthetic and real-world datasets, we show that our framework delivers more accurate forecasts than the state-of-the-art models, depicting, in which quantiles, the intervention most affected the treated units, unlike the conventional counterfactual inference methods based on nonprobabilistic approaches.**

Manuscript received 22 October 2021; revised 24 April 2022; accepted 6 July 2022. This work was supported in part by the Australian Research Council under Grant DE190100045, in part by the Monash University Graduate Research Funding, and in part by the MASSIVE High Performance Computing Facility, Australia. *(Corresponding author: Priscila Grecov.)*

Priscila Grecov, Ankitha Nandipura Prasanna, and Christoph Bergmeir are with the Department of Data Science and Artificial Intelligence, Monash University, Melbourne, VIC 3800, Australia (e-mail: priscila.grecov@monash.edu).

Klaus Ackermann is with the Department of Econometrics and Business Statistics, Monash University, Melbourne, VIC 3800, Australia.

Sam Campbell, Debbie Scott, and Dan I. Lubman are with Turning Point, Eastern Health and Eastern Health Clinical School, Monash University, Melbourne, VIC 3800, Australia.

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TNNLS.2022.3190984.

Digital Object Identifier 10.1109/TNNLS.2022.3190984

***Index Terms*— Causal effect, counterfactual analysis, global time series forecasting, neural networks (NNs), probabilistic forecasting.**

I. Introduction

THE problem of estimating the causal effects of a treatment or policy intervention over a time series of interest spans across numerous areas from public policy and governance to commercial applications. Traditional A/B testing or randomized control trials are either not ethically possible or commercially feasible. In this context, there is a need for interpretable predictions using observational data. Causal statements resulting from an intervention usually rely on forecasting counterfactual time series outcomes based on similar units unaffected by the intervention at the same point in time. This artificial counterfactual trend is then used to estimate the causal effect of the intervention by differentiating the observed series affected by the intervention and the artificial counterfactual. These comparative case studies are ubiquitous in empirical research in the social sciences with different approaches for estimating the causal effects of an intervention. Most econometric tools to perform this task focus on predicting the counterfactual time series from structural causal models. Deep learning (DL) forecasting techniques have evolved at a slower pace in this domain. Examples of the econometric current approaches for estimating the policy causal effects include the difference-in-difference (DiD) approaches [1]–[3], synthetic control method (SCM) [4], matrix completion [5], factor and interactive fixed effect (FE) models [6], constrained LASSO regression-based methods [3], [7], [8], and Bayesian structural time series models [9]. All these methods, which are called in the causal inference literature "counterfactual and SCMs," follow the original idea of the Rubin causal model [10], which is also called the *potential outcome* framework, where the most popular approach is the SCM.

Recently, in forecasting, classical forecasting techniques, such as ARIMA [11] and exponential smoothing models [12], have been increasingly outperformed and replaced by DL techniques, which can learn more adequately complex patterns across a collection of several time series delivering more accurate predictions [13], [14]. DL techniques have superior performance with many related time series, multiple covariates, and complex nonlinear relationships between inputs and outputs. This important transition in the time series forecasting domain is closely related to the change from *local*, per-series

univariate modeling to *global* forecasting models (GFMs), which learns across many time series [14]–[16]. The idea behind the *global* approach is to pool all time series in the dataset together and train a single forecasting model across all time series. Hence, the *global* approach enables us to overcome the problem of limited availability of data in *local* models and, thus, avoids overfitting and enables us to train more complex models on larger volumes of data (all time series versus a single time series). While many studies state a requirement for "relatedness" across time series [14], [17], recent works have shown theoretical justifications and have demonstrated good performance for GFMs even when the set of time series cannot be considered related [13], [16].

Estimating the entire probability distribution of a time series' future values conditioned on its past, rather than just producing a point estimate, is known in the literature as probabilistic forecasting. It is becoming increasingly important, as it allows for decision-making under uncertainty. Being capable of precisely predicting intervals to quantify forecast uncertainties is often crucial for estimating the risks associated with decision-making processes. Furthermore, the analysis of target interventions is a key input for optimizing business processes. However, the difficulty of applying classical probabilistic forecasting methods to real-world datasets is typical that they assume Gaussian distributions for the observed data, which is often not realistic. To overcome this difficulty, Gasthaus *et al.* [18] proposed a methodology that combines recurrent neural networks (RNNs) with the flexibility of a quantile function-based specification of the observations' distribution, which does not require parametric assumptions made on the output distribution.

Within the realm of causal effect estimation, there is a large body of literature focusing on econometric methods associated with structural models, and recently, some works also introduce counterfactual prediction methods that are based on deep neural networks (DNNs) [19]–[24]. Those studies argue that the nonparametric nature of DNN models overcomes the nonlinear, nonconvex limitations of traditional structural counterfactual prediction models. Nonetheless, the underlying forecasting methods used in these DNN-based counterfactual prediction frameworks are mostly *local* models, in contrast to the more recent state-of-the-art time series forecasting methods that are *global* models. Our earlier work in [25] is, to the best of our knowledge, the only work, thus far, which started to employ GFM-DNN-based counterfactual predictors to perform causal impact analysis under the SCM approach, albeit without considering the probabilistic forecasting aspect.

Consequently, we fill the gaps mentioned thus far and reap all the benefits that GFM-DNN and quantile-probabilistic time series forecasting techniques bring with them to the causal inference domain. We present the *Deep Probabilistic Counterfactual Prediction Net* (DeepProbCP), a novel counterfactual probabilistic forecasting framework based on autoregressive RNNs with conditional quantile functions, which learns a *global* model from historical preintervention data of *all treated and control time series* in the dataset, to estimate the causal effect produced by policy interventions in a group of several treated units. This multiple-composed-technique framework is capable of handling the challenges present when training on multiple time series jointly in real-world forecasting problems with wide distinguished magnitudes, nonlinear relationships, and strongly skewed or heavy-tailed non-Gaussian distributions, which are typical in practical applications.

A. Contributions of This Article

We address counterfactual prediction for causal effect analysis when it is not possible to assume: long-term dependencies with linear and convex relationships between the inputs and outputs, absence of complex seasonal patterns and trends, uniform interventions, and distribution equivalence features (and, therefore, a notion of similarity between the treated and control units in the preintervention period) for an observed nonlinear and high-dimensional [26] set of treated and untreated time series with multiple treated units. We introduce a method to evaluate intervention effects estimated by SCMs that, in contrast to conventional approaches, uses a *global* probabilistic DNN forecasting method as the estimator model to predict the counterfactual trend and distribution outcomes for the treated units. Our approach is especially useful in high-dimensional data situations where there are multiple treated units, the relationships between the control and treated time series are complex, and the intervention affects only specific quantiles of the treated units' distribution, allowing more realistic modeling. Furthermore, our approach is robust to the presence of confounding effects.

The identification of the intervention effect relies on the general assumptions needed to forecast time series. We pool univariate time series to build one model with shared parameters for all pretreatment time series. Next, we use the shared parameters to forecast each univariate time series in control and treatment posttreatment. For the treated time series, the forecasts act as an estimation of the counterfactual, how the series would have behaved without treatment. On the other hand, for the control time series, as we are using the same model, the forecasts test to verify that our model can still provide reliable forecasts during the posttreatment period, increasing our confidence in the estimates of the treated units.

To the best of our knowledge, this is the first study that employs a GFM combined with DNN models and quantile probabilistic forecasting to conduct causal effect analysis using the SCM approach. We propose a general counterfactual modeling framework that employs a relatively new field of *global* modeling for counterfactual time series prediction, initially introduced by us in [25], now combining it with quantile probabilistic forecasting to build a spline-based representation of the counterfactual distribution capable of measuring the causal effect of policy interventions affecting only parts of the distribution of the treated units. Compared to classic SCM approaches, DNN-probabilistic GFMs are better suited for making counterfactual predictions as they learn across multiple time series simultaneously and forecast the uncertainty of the outcome as probability distributions and use those forecasts to predict the causality. Several decision-making situations require richer information available from a probabilistic forecasting model that returns the full conditional distribution, rather than a point forecast model that predicts only, for

example, the conditional mean. Traditional probabilistic forecasting is often implemented assuming a Gaussian error distribution on the residual series. However, this exact parametric distribution is often not observed in real-world applications. Therefore, particular quantiles of the forecast distribution are useful in making optimal decisions, leading to the use of quantile regression (QR) combined with probabilistic RNN models to generate multihorizon quantile forecasts. In QR, models are trained to minimize quantile errors, which is also known as quantile-loss or *pinball-loss* functions. QR is robust because it does not ask for distributional assumptions and returns accurate probabilistic forecasts with sharp prediction intervals [27], [28].

In *local* and classic SCM approaches, training during the pretreatment period takes into consideration only the combination of control units or external predictor time series as covariates and then transfers the learned parameters (e.g., network weights) to the treated time series in the postintervention period to predict the counterfactual. With a *global* approach, when we add the treated unit before the intervention effect to the training phase, we can add more information to the modeling, and therewith, the counterfactual forecasting is enhanced with more data, which consequently produces more accurate causal inference. Furthermore, we neither need to make an assumption of equivalence between distributions of the control and treated outcomes in the pretreatment period nor need to search for or limit ourselves to similar control units (therewith also not having to define a notion of similarity) as the classic SCMs do [25]. With a GFM approach, we are in addition reducing the risk to bias the results by presplitting our data in the control and treated groups.

Conceivable interventions that might affect the forecast are not handled by any state-of-the-art probabilistic forecasting methods. In our earlier work on DeepCPNet in [25], the counterfactual approach of predicting a causal effect is obtained as the full distribution in which the most affected quantiles of the predicted counterfactual distribution from the policy intervention are not well understood. Both of these limitations are addressed by our proposal when combining both the QR approach for probabilistic forecasting and the DeepCPNet counterfactual approach for predicting the causal effect. Our DeepProbCP framework incorporates state-of-the-art methods of forecasting quantiles using the quantile loss (QL) function for probabilistic global forecasting methods, as in [18] and [28], and DeepCPNet [25] as the counterfactual prediction.

Finally, it is clear that using QR as probabilistic forecasting provides more accurate quantiles by observing all available time series globally. The quantile forecasts obtained from QR are further used to estimate the counterfactual distribution and, consequently, to measure the impacts caused by the policy intervention.

We demonstrate in detail how to combine probabilistic methods with the DeepCPNet framework to achieve better performance than state-of-the-art models in simulated studies considering a complex and nonlinear data generating process (DGP) and in forecasting applications in real-world datasets. The empirical experiment evaluates the impact of the growth in the number of alcohol licenses issued (ALI) on emergency medical services (EMSs) demand attendances related to alcohol intoxication reported in Australia.

B. Outline of This Article

In Section II, we discuss the prior work. In Section III, we present the architecture of our proposed DeepProbCP framework in detail. In Section IV, we describe the simulated study and an empirical exercise over a real-world dataset. In Section V, we analyze the results obtained from the experiments and discuss the main insights resulting from them, demonstrating the performance, as well as the value and practical usability of DeepProbCP. Finally, Section VI concludes this work.

II. Related Work

In the following, we revisit the main literature from the field of causal inference, as well as the relevant background literature from the neural network (NN) and probabilistic forecasting fields.

A. Structural Causal Models Under the Counterfactual and Synthetic Control Approach

Causal inference refers to an intellectual discipline that considers the assumptions, study designs, and estimation strategies that allow researchers to estimate the effects of causal variables (which is also referred to as treatment) on some outcome of interest. Active dedicated research on causality dates back to the 1980s [29]. Recently, Athey and Imbens [30] reviewed several strategies to estimate the causal impacts of observational data using the counterfactual and synthetic control approach.

The limitations of DiD estimation [1]–[3] rely on a linear parallel trend hypothesis between the control and treated units. The assumption is that the level of the control unit group can be an adequate proxy for the level of the treated unit group in the absence of the intervention. Gobillon and Magnac [6] generalized the DiD approach by estimating a correctly specified linear panel model with strictly exogenous regressors and interactive FEs. However, this approach requires the model to be correctly specified, estimates the common factors, and requires the regressors to be strictly exogenous, which is a strong assumption in most applications with aggregate time series data. Although SCM addresses many of these limitations, this estimator relies on the strong assumption of a convex combination of covariates to construct the counterfactual, which biases the estimator [7], [31]. This convexity weight restriction also precludes nonconvex and nonlinear relationships between the multiple unaffected units (control units), assuming a very particular fit by the model in the pretreatment period. Furthermore, the weights in the SCM are usually estimated using the time averages of the observed variables for each peer, which removes all the time series dynamics. Finally, the SCM does not provide any guidance on how to select the variables that determine the optimal weights. Doudchenko and Imbens [3] and Carvalho *et al.* [7] explored alternative methods for calculating appropriate weights for an SCM approach, such as best subset regression (penalized regression) or least absolute shrinkage and selection operator (LASSO) and elastic net methods, which performs better

in settings with a large number of potential control units. Finally, Brodersen *et al.* [9] introduced an alternative to SCM proposing the use of state-space models, more specifically structural time series models, combined with flexible regression components (relaxing the convexity restriction) to forecast the temporal evolution of an observed outcome.

However, all these structural causal models build the counterfactual as a function of observed variables from a pool of covariates (control units and/or external predictors). Therefore, the key assumption for these approaches is that the relationship between the covariates $\mathbf{x}_1, \ldots, \mathbf{x}_t$ and each treated time series $y_1, \ldots, y_t$ modeled during the preintervention period remains the same postintervention as follows:

$$y_t = \alpha_0 + \beta_t' \mathbf{x}_t + \varepsilon_t \quad \forall t = 1, \ldots, T_0 \tag{1a}$$

$$\hat{y}_t = \alpha_1 + \hat{\beta}_t{}' \mathbf{x}_t + \zeta_t \quad \forall t = T_0 + 1, \ldots, T \tag{1b}$$

where T_0 is the intervention time, and the error terms ε_t and ζ_t are independent identically distributed (i.i.d.). Based on this, the fit model in (1a) is used to estimate the counterfactual time series for treated units postintervention, as in (1b). The current structural causal models based on the SCM approach must assume that the posttreatment relationship between the treated and predictor time series is the same as that mapped prior to the intervention to enable the application of the estimated parameters over the posttreatment predictor time series and, therefore, allow for identification. As we will see next, our proposed framework does not need to assume this strict dependence because our estimation process is based only on the pretreatment data. Another limitation of structural models is their parametric nature. The use of nonparametric models, such as DNN models, adds more flexibility to the modeling process because the functional relationship between the entities does not need to be specified.

B. Neural Network Models for Counterfactual Prediction and the Global Forecasting Method

Recently, DNN-based counterfactual prediction methods have been introduced [19]–[24], [32] in the causal inference literature. These studies argue that the nonparametric nature of DNN models obviates the nonlinear, nonconvex limitations of SCMs for counterfactual prediction. Considering the available complex datasets, the powerful activation functions and optimization techniques present in the NNs improve the prediction accuracy of the counterfactual outcomes, as demonstrated by Poulos [24] and Steinkraus [32]. Nonetheless, the underlying DNN forecasting methods used in current counterfactual prediction frameworks are mostly *local* models. In contrast, state-of-the-art time series forecasting has moved from per-series *local* modeling to GFMs that learn across many time series [14], [33], [34]. The term *global model* was introduced by Januschowski *et al.* [15]. They differentiated the methods that estimate the parameters independently for each time series (*local* models) and in conjunction with all available time series (*global* models). The *global* or *cross-learning* methodology pools all time series of the dataset together and fits a unique forecasting function (consequently, with a loss function being computed once for all time series and weights being learned globally across all series in the dataset). Therewith, *global* forecasting frameworks exploit the rich cross-series information available during the learning process, which is not possible for *local* models.

Montero-Manso and Hyndman [16] demonstrated in their work the underlying formal principles that justify the better accuracy of *global* methods. First, those authors show that the forecasts produced by the many functions of a *local* method (one per series) can always be replicated by a single (complex enough) function learned by a *global* algorithm. Second, they demonstrate that the complexity of *local* algorithms increases as the sample size increases, surpassing the constant complexity of *global* algorithms, which does not increase with the size of the sample set once the number of parameters of its single forecasting function remains the same. Therefore, for the same sample set size, *global* models have space to be more complex (by adding features or increasing model order, for example) than *local* models and consequently deliver more accuracy and better generalization.

Grecov *et al.* [25] extended the GFM-RNN framework presented by Bandara *et al.* [33] to perform causal effect analysis by introducing DeepCPNet, the first GFM-RNN-based method for counterfactual estimation. The GFM approach reduces the risk of bias in the results by presplitting the data into control and treated groups. It can also relax the usual causal model assumptions of distribution equivalence between the control and treated time series, as well as the assumption that this relationship observed during the pretreatment period remains the same postintervention. A long short-term memory (LSTM) network is used to model the nonlinear, nonconvex, and dynamic interactions between the treated and control groups. This architecture is capable of capturing sequential long-term dependencies and alleviating gradient vanishing/exploding problems. This approach is a promising framework for counterfactual prediction for causal impact evaluation from a *global* modeling perspective.

C. Probabilistic Forecasting Methods

The transition from point estimation to distribution estimation has been explained by Stigler [35] and addressed by Gneiting [36], who elucidated the importance of the loss function in forecasting practices that lead to optimal point predictors of a simple analytical form, equal to the quantile of the predictive distribution. According to Gneiting and Katzfuss [37], probabilistic forecasting methods forecast the future in the form of a probability distribution and aim to maximize the sharpness of the forecast distribution based on the past available information set.

In this field, Salinas *et al.* [17] proposed the DeepAR model that allows the model to learn the seasonal behaviors and dependencies on covariates across time series and expectantly produces the full probability distribution of the forecasts, which is vital for optimal decision-making. It incorporates the negative binomial likelihood for count data and special treatments when the magnitudes vary. The architecture of DeepAR uses a Bayesian RNN framework to make the system deterministic, nonlinear, and dynamic, whereas the gradient vanishing and exploding are handled by the use of LSTM

nets. DeepAR is very effective in learning *global* models and is able to learn complex patterns of seasonality and uncertainty over time. However, this method requires previous parametric assumptions made on the output distribution. Most techniques used in probabilistic modeling assume a Gaussian distribution of the outputs, which is mostly done for mathematical reasons rather than because of empirical evidence.

Following the Gneiting [36] method, Wen *et al.* [28] proposed a framework for probabilistic multistep time series regression that outperforms DeepAR without any parametric assumptions about the distributions. Their framework implements an efficient scheme of sequence-to-sequence (Seq2Seq) NN to model the nonparametric nature of QR combined with a multihorizon forecast. Instead of a full distribution, only certain quantiles are obtained. In [28], the model learns the parameters by minimizing a pinball loss function.

Gasthaus *et al.* [18] also proposed a nonparametric RNN-based probabilistic forecasting framework, which makes use of a spline-based specification to forecast the quantile distribution. The training procedure of this NN for learning the model parameters is based on the minimization of the continuous ranked probability score (CRPS) (instead of the QL), where conditional quantile functions are estimated by using regression splines. This framework, initially proposed for probabilistic forecasting, is partially adopted in this article, being used to the best of our knowledge for the first time for causal effect estimation under the SCM approach. A major advantage of this probabilistic modeling strategy (using QR and splines) is that it does not require the imposition of a previous statistical distribution, leading to a more realistic and robust approach than probabilistic methods that require parametric assumptions. Moreover, Gasthaus *et al.* [18] addressed the issue of quantile crossing, which is a limitation of the Wen *et al.* [28] work.

However, these state-of-the-art methods of quantile probabilistic forecasting have not been employed yet in the causal inference domain. The forecasting procedure to predict the intervals or the full probability distribution of the prediction does not *a priori* deal with the causal effect of any policy intervention that occurs in the data. In the works of Gasthaus *et al.* [18] and Wen *et al.* [28] of multihorizon quantile recurrent forecasters, each quantile can be fit to the model simultaneously, and multiple quantile outputs can be obtained. Even though the most accurate quantiles are predicted, there still is no evaluation of causation.

Motivated by this gap, we introduce a novel methodology for probabilistic time series forecasting for counterfactual prediction, combining the GFM-RNN-based DeepCPNet framework for counterfactual prediction with techniques associated with quantile probabilistic modeling using splines, as proposed by Gasthaus *et al.* [18], in order to predict the distribution of the counterfactuals. Hence, adopting a quantile-spline probabilistic framework where the counterfactual prediction is not confined to a point estimation but to a distribution estimation, our method models uncertainty in causal effect inference, which is crucial for guiding relevant decisions.

III. Method

In the following, we first define the counterfactual and causal effect estimation problem for counterfactual and SCMs. Second, we introduce the GFM-RNN-based forecast engine used to generate the counterfactual predictions in the DeepProbCP framework. We close the section by presenting how we incorporate the quantile-spline probabilistic technique into the previous method to forecast the counterfactual full distribution.

A. Counterfactual and Causal Effect Problem

Let us consider an observed set of time series with J units (for example, firms, products, states, and countries) indexed by $j = 1, \ldots, J$, and let T be the time period indexed by $t = 1, \ldots, T_0, \ldots, T$ with $T_0 < T$. Furthermore, we assume that an intervention occurs in a subset $i = 1, \ldots, I$ of units ($n(i) > 0$ and $I < J$) at time T_0. Thus, these treated units i are not affected by the policy intervention during the first T_0 periods but are treated by the policy during $T - T_0 = T_*$. In the counterfactual problem, T_* is typically shorter than the period from the start to T_0. In this group of time series, there is also additional information in the form of several untreated units (not affected by the policy intervention), which serve as control instances—the control units. These control units are the remaining units $j \neq i$ unaffected by the policy for all time periods T and indexed by $c = 1, \ldots, C$ ($C < J$). The idea is to forecast an artificial counterfactual outcome for the treated units based on this high-dimensional set of observed control and treated time series using the information prior to the intervention period T_0. This artificial counterfactual is then used to estimate the treatment effect (TE) of an intervention, which is the main goal of causal effect estimation.

Hence, the TE estimation becomes the application of a forecast model to generate mean-unbiased proxies, $P_{i,t}^{(0)} := \mathcal{M}(Y_{j,t}^{(0)}; \theta)$, where $\mathcal{M}(\cdot, \cdot)$ represents a general mapping function taking into account all true time series values $Y_{j,t}^{(0)}$ and model parameters θ, for producing the counterfactual of i treated units in the absence of the policy intervention, $Y_{i,t}^{(0)}$, the true counterfactual, in the posttreatment period ($t > T_0$) as follows:

$$Y_{i,t>T_0}^{(0)} = P_{i,t>T_0}^{(0)} + u_t, \quad E(u_t) = 0, \ t = 1, \ldots, T_0 + T_* \quad (2)$$

which, under the assumption of stationarity on $Y_{j,t}^{(0)}$ and that the exact timing of the start of the intervention is known, leads us to identify the TE of a policy in period t as

$$\widehat{\delta}_t := Y_{i,t>T_0}^{(1)} - Y_{i,t>T_0}^{(0)} \quad (3)$$

where $Y_{i,t>T_0}^{(1)}$ is the observed outcome of the treated units affected by policy intervention, and we are interested in the prediction of policy effects only for the postintervention period ($t > T_0$).

The estimation of the *total TE* (TTE) for the time that the policy intervention takes place is calculated by $\widehat{\text{TTE}} := \sum_{t=T_0+1}^{T} \widehat{\delta}_t$, and the *average TE* (ATE) on the treated units

can be represented by

$$\widehat{\Delta}_T = \frac{1}{T - T_0} \sum_{t=T_0+1}^{T} \widehat{\delta}_t. \tag{4}$$

The observed outcomes of the control units are denoted by $Y_{c,t}^{(1)}$. Then, in contrast to the treated units, the TE δ over the control units is supposed to be null. Thus, the SCM approach assumes that the observed outcomes of the control units are equal to their outcomes in the absence of the intervention, that is, $Y_{c,t}^{(1)} = Y_{c,t}^{(0)}$ for $1 \leq t \leq T$, which is different from what is specified in (3) for the treated units.

Given these definitions, the SCM approach considers the following two essential assumptions, as described by Chernozhukov *et al.* [8].

Assumption 1 (Counterfactual Estimation Using the Potential Outcome Approach): Let $P_{i,t}^{(0)}$ be unbiased predictions for the counterfactual outcomes for treated units $Y_{i,t}^{(0)}$ in the absence of the treatment, that is, $E(P_{i,t}^{(0)}) = E(Y_{i,t}^{(0)})$, and $\delta_t = 0$ for $t \leq T_0$. Then, the treated potential outcomes under the effect of the intervention ($t > T_0$) are given by $Y_{i,t}^{(1)} = Y_{i,t}^{(0)} + \delta_t$ and can be expressed as follows:

$$\left.\begin{aligned} Y_{i,t}^{(0)} &= P_{i,t}^{(0)} + u_t \\ Y_{i,t}^{(1)} &= P_{i,t}^{(0)} + \delta_t + u_t \end{aligned}\right|, \quad \text{with } E(u_t) = 0,\ t = 1, \ldots, T.$$

Here, u_t is a stationary stochastic process, and the relation between the observed outcomes and potential outcomes is given by $Y_{i,t} = Y_{i,t}^{(0)} + D_t(Y_{i,t}^{(1)} - Y_{i,t}^{(0)})$, where $D_t \in (0, 1)$ is a binary variable that flags the preintervention or postintervention period, which means that $D_t = 1$ when $t > T_0$, and $D_t = 0$ otherwise.

Assumption 2 (Null Intervention Effect Over Control Units): $Y_{c,t}$ is independent of D_t for all t, that is, $Y_{c,t}^{(1)} = Y_{c,t}^{(0)}$ for $1 \leq t \leq T$.

Assumption 1 introduces the potential outcomes postulating the existence of unbiased proxies for $P_{i,t}^{(0)}$ without imposing restrictions on their dependence on u_t. It requires u_t to be i.i.d. or at least a stationary and weakly dependent process. It also assumes that the stochastic shock u_t is invariant under the intervention, which means that the timing of the intervention should be independent of instances that influence the u_t distribution.

Assumption 2 is sufficient for the control units to be unaffected by the intervention. This assumption is necessary to recover the effects of the intervention. In contrast to the usual SCM approach, this assumption fulfills a different goal for us. Given that $Y_{c,t}$ is independent of D_t, we utilize *global* models with shared parameters θ across all series j, and the prediction of $Y_{c,t}$ serves as a check of being able to forecast in the posttreatment period $t > T_0$. θ is estimated in the pretreatment period and is used for predicting the control and treated time series. Therefore, when we test for the null effect, implicitly, this is a test of how well the model parameters perform in the posttreatment period.

B. DeepProbCP Probabilistic Forecast Framework

In this section, we describe the architecture of our proposed method GFM-RNN, the loss function, how the network is trained, the preprocessing considerations in the input feature process, and, finally, the spline construction of our predictive distribution using the quantile forecasts. The DeepProbCP framework is a two-step procedure. In the first step, the pretreatment data are used to estimate the defined Q quantiles of future trajectories for each time series—the counterfactuals in the case of the treated units. In the second step, the predicted counterfactual distribution is built by linking the quantile forecasts via spline functions.

Formally, we recast the causal effect estimation problem as a counterfactual forecasting problem to estimate more reliable $P_{i,t}^{(0)}$. There are several modeling strategies for estimating the counterfactual proxies $P_{i,t}^{(0)}$. In our framework, we focus on the application of GFM-RNN forecast modeling to generate mean-unbiased counterfactual proxies, $P_{i,t}^{(0)}$. Because we employ a *global* modeling approach instead of structural causal models, $P_{i,t}^{(0)}$ can be estimated considering both treated and control time series in the pretreatment data, $Y_{i,t \leq T_0}$ and $Y_{c,t \leq T_0}$, as input variables to train the model—the covariates. Therefore, assuming, without loss of generality, that $Y_{j,t} \in \mathcal{Z} \subseteq \mathbb{R}^d, d > 0$, and $Y_{i,t}^{(0)} \in \mathcal{Y} \subseteq \mathbb{R}^q,\ q > 0$

$$Y_{i,t>T_0}^{(0)} = \mathcal{M}(Y_{j,t \leq T_0}, \theta) + u_t \tag{5}$$

where $\mathcal{M} : \mathcal{Z} \times \Theta \to \mathcal{Y}$ is a measurable mapping (the model) for each $\theta \in \Theta$, a finite-dimensional parametric space.

The SCM requires the choice of a model $\mathcal{M}(\cdot, \cdot)$, which captures the information from the treated and control units using their preintervention data, which means that the counterfactual is built considering a model trained in the absence of the intervention. Instead of using classical econometric causal models, such as DiD, SCM, penalized and constrained Lasso regression estimators, and Bayesian time series models (such as the CausalImpact framework [9]), we propose the use of a more flexible model (without the strong structural causal model assumptions) by the application of RNNs for autoregressive time series forecasting in the context of the global forecasting approach combined with quantile probabilistic forecasting techniques to estimate the counterfactual distribution as an outcome.

1) Network Model—LSTM: Our approach for the network modeling is based on an approach established in the literature by Grecov *et al.* [25] and others [14], [34]. In particular, we add to that framework the quantile probabilistic forecasting, replacing the mean absolute error (MAE) loss function with the QL function during the learning and hyperparameter tuning processes.

Considering a set of J univariate time series (our treated and control units as pointed out in Section III-A.) $\{y_{j,1:T_0}\}_{j=1}^{J}$, the objective of our network is to model the future trajectories of each time series $\{y_{j,T_0+1:T}\}_{j=1}^{J}$ after the intervention given its past preintervention data (in the absence of the intervention), where the preintervention data from both the treated and control units, together with a set of associate external covariate vectors $\{\mathbf{x}_{j,1:T_0}\}_{j=1}^{J}$, are the inputs/covariates of our global network as follows:

$$\{y_{j,T_0+1:T}\}_{j=1}^{J} = m_G(y_{:j,1:T_0}, \mathbf{x}_{:j,1:T_0}, \Theta) \tag{6}$$

where m_G denotes our *global* network model, Θ denotes the set of learnable parameters of the network model, $y_{j,t} \in \mathbb{R}$ is the value of the jth time series at time t, and $\mathbf{x}_{j,t} \in \mathbb{R}^D$ is an external covariate vector.

Thus, at a high level, our modeling setup is as follows: the preintervention time series $y_{j,t}$ and the additional (optional) external covariates $\mathbf{x}_{j,t}$ are provided to an autoregressive LSTM-based whose architecture follows that described by Grecov *et al.* [25] and others [14], [34]. This LSTM is autoregressive and recurrent in the sense that it takes as inputs the values of the previous time step and previous state of the network. Hence, this autoregressive LSTM $m_G(\cdot)$ model can be expressed in terms of its hidden states (h_t) as follows:

$$\mathbf{h}_{j,t} = r(\mathbf{h}_{j,t-1}, y_{:j,t-1}, \mathbf{x}_{:j,1:T_0}, \Theta). \tag{7}$$

In (7), the function $r(\cdot)$ is a multilayer RNN with LSTM cells (parameterized by Θ) with peephole connections using the stacked LSTM design, as recommended by Bandara *et al.* [14] and Hewamalage *et al.* [34] to accommodate our *global* approach. Because implementation follows the GFM approach, the set of parameters Θ are learned globally across all J time series, which adds much more information to our modeling, consequently boosting the accuracy of our counterfactual forecasting.

2) Quantile Loss: QL is a method that enables quantile forecasting. In quantile forecasting, the loss function is also called *pinball loss*. If we let τ be an arbitrarily chosen quantile, y the observed value, $\hat{y}$ the quantile forecast, and $(\cdot)_+ = \max(0, \cdot)$, we describe the QL as

$$L_\tau(y, \hat{y}) = \tau(y - \hat{y})_+ + (1 - \tau)(\hat{y} - y)_+. \tag{8}$$

We use the QL with τ as the quantile we want to forecast, and a GFM-LSTM (as described in the previous subsection) as the underlying network model $\mathcal{M}(\cdot, \cdot)$. When $\tau = 0.5$, the QL is simply the MAE, minimized by the median of the forecast distribution. Let K be the length of the forecast horizon, Q the number of quantiles, and J the quantity of time series. Then, the $K \times Q \times J$ tensor $\hat{\mathbf{Y}} = \left[\hat{y}^{(\tau)}_{:j,t+k}\right]_{k,\tau,j}$ is the output of the model $\mathcal{M}(\cdot, \cdot)$. The DeepProbCP method of probabilistic modeling consists of forecasting a wide range of prescribed quantiles (following the method proposed by Wen *et al.* [28]), for all time series j, training a separate *global* LSTM per each quantile. A full list of quantiles is then assembled by using spline interpolation to model the quantile distribution (as explained in Section III-B4.) as follows:

$$\begin{cases} \left(\hat{y}^{(\tau_1)}_{:j,t+1}, \ldots, \hat{y}^{(\tau_1)}_{:j,t+K}\right) = m_G\left(h_t, x^{(h)}_{:j,:t}, \tau_1; \Theta\right) \\ \left(\hat{y}^{(\tau_2)}_{:j,t+1}, \ldots, \hat{y}^{(\tau_2)}_{:j,t+K}\right) = m_G\left(h_t, x^{(h)}_{:j,:t}, \tau_2; \Theta\right) \\ \ldots \\ \left(\hat{y}^{(\tau_Q)}_{:j,t+1}, \ldots, \hat{y}^{(\tau_Q)}_{:j,t+K}\right) = m_G\left(h_t, x^{(h)}_{:j,:t}, \tau_Q; \Theta\right) \end{cases} \tag{9}$$

where $y_{:j,:}$ are all the j time series to forecast, m_G is our *global* LSTM with its parameters shared across all horizons $k \in \{1, \ldots, K\}$ and unit time series $j \in \{1, \ldots, J\}$, h_t are the hidden states of our LSTM where all history is encoded, $\tau_{(\cdot)}$ denotes each of the Q quantiles, and $x^{(h)}_{:j,:t}$ are the temporal inputs/covariates available in the preintervention history $[Y_{j,t \le T_0}$ from (5)]. For simplicity, we are not including the vector of external covariates separately in (9), but it could be added. The series are fed into one LSTM model sequentially, thus enabling cross-series learning.

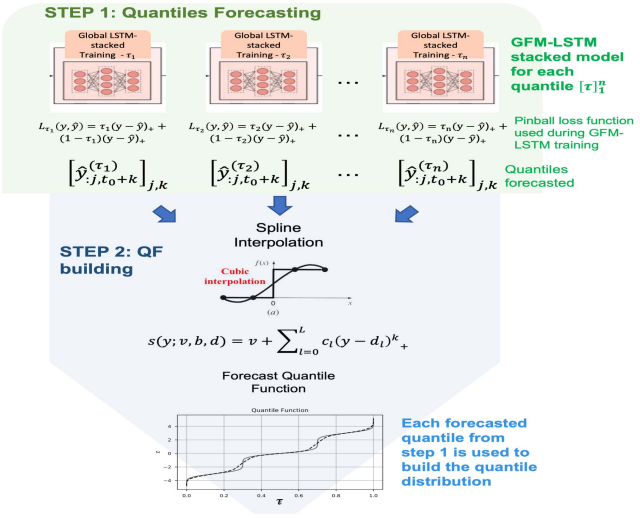

Fig. 1. Design of DeepProbCP framework: Step 1 (forecasting of the quantiles) and Step 2 (building the quantile function) of DeepProbCP.

The system in (9) can be simplified by $\hat{\mathbf{Y}} = m_G(h_t, x^{(h)}_{:j,:t}, \tau; \Theta)$, where $\hat{\mathbf{Y}}_i$ corresponds to the Q counterfactual quantile predictions for all treated units, whereas $\hat{\mathbf{Y}}_c$ is the Q quantile forecast for all control units. Fig. 1 summarizes the working of DeepProbCP.

3) Preprocessing and Training: To train the DeepProbCP, we split each time series into fixed-size windows with different starting points using the past observations of time series in the form of input and output windows, thereby generating multiple training instances, which is known as the moving window transformation strategy. For each window, we compute the QL as in (8). This loss is then minimized using COntinuous COin Betting (COCOB) as the primary learning algorithm (see [34] for further details). From each time series, we reserve the last output window of the training dataset as the validation set for the automatic hyperparameter tuning. The last output window of the entire dataset is reserved for testing the accuracy of our forecasts. To perform this hyperparameter tuning, we use the sequential model-based configuration (SMAC) algorithm that implements a Bayesian hyperparameter optimization tuning process [34].

Furthermore, we employ the seasonal exogenous (SE) training approach proposed by Bandara *et al.* [38] to train our framework. In the SE approach, the extracted seasonal components are used as exogenous variables in the original time series observations. This step supplements DeepProbCP to learn the seasonality present across multiple related time series. Typically, in real-world time series datasets, the series are all aligned and commonly come all from similar sources with similar characteristics, such as the shape of the seasonal pattern or trend. Thus, depending on individual series lengths and potentially other characteristics, it can be beneficial to learn seasonality and trend across the series or per series. Furthermore, seasonality and trend can be tackled by a

preprocessing step as proposed, e.g., by Smyl [13] for the M4 dataset and further explored by Bandara *et al.* [38], so that the RNN works on deseasonalized, detrended series. As our series are relatively homogeneous and short, seasonality and trend estimation per series may be unnecessarily difficult, and we opt for an algorithm that exposes the full input series to the RNN while giving it additional information on the patterns of the individual series in the form of additional inputs.

Finally, a preprocessing layer is applied before the training layer. In this step, the time series is first normalized using mean-scale transformation. Second, we apply a log transformation to stabilize the variance across the series, which also ensures that seasonality and trends in the series are additive. Finally, we extract the seasonal components from all series, using a decomposition technique, to use them as an exogenous variable during the training, as mentioned previously. After the training layer, we invert the transformations applied during preprocessing for the final outputs. We implement the DeepProbCP framework using the open-source deep-learning toolkit TensorFlow, version 2.0 [34].

4) Splines: Once multiple quantiles are calculated, a method is still needed to model the full form of the quantile function against the domain [0, 1] for each time series and forecast data point.

We perform quantile distribution modeling using spline interpolation. Splines are singular functions defined by piecewise polynomials that match given values (or restrictions) at certain chosen points $\{y_1, \ldots, y_t\}$. By using splines, we can model our quantile distribution by enforcing that certain points, such as quantiles, will be connected in a smooth fashion. The enforcement of smoothness in splines is made through knots that connect diverse polynomial functions and avoid Runge's phenomenon [39]. Spline interpolation is preferred over polynomial interpolation because it can achieve high interpolation performance even when using low polynomial degrees while avoiding high variance when using higher degrees. The spline functions are described as follows:

$$s(y; \upsilon, b, d) = \upsilon + \sum_{l=0}^{L} b_l (y - d_l)_+^k \tag{10}$$

where υ is the intercept term, b is a vector of weights that describe the slopes of each function, k is the degree of smoothing of the spline, d is a vector of knot positions, the number of pieces L is a hyperparameter of the spline, and $(\cdot)_+ = \max(0, \cdot)$ is the "hinge" or rectified linear unit (ReLU) function. After calculating a range of quantiles for each forecast time point and time series, we run individual (cubic, $k = 3$) spline regressions with the `scipy` Python package [40], calculating the number of knots as a function of the positive smoothing factor. We highlight that this is an independent step from quantile forecasting (in contrast to the approach of Gasthaus *et al.* [18]), and every forecast point requires a spline regression function to model.

IV. Experiments

In this section, we first describe a simulation experiment designed to analyze the functioning of the proposed method. Then, we present an empirical study with the description of the dataset used, the baseline models selected to challenge our method, and the error metrics employed to evaluate the performance of the models. Finally, we discuss the statistical significance (hypothesis testing) and placebo tests that we perform.

A. Simulation Study

Since counterfactuals are unobservable, it is difficult to determine which estimates are closer to the true TEs. Therefore, as the first experiment, we propose to use synthetic evidence to compare which estimates could be closer to the actual TEs by conducting computer simulations.

We built 24 simulated scenarios by a combination of linear and nonlinear DGPs, homogeneous and heterogeneous TEs (i.e., the intervention affecting treated units in a uniform and nonuniform way, respectively), two different time series lengths (60—short and 222—long), and three different amounts of time series (10, 101, and 500). For the linear DGP, we generate stationary series from autoregressive AR models, and for the nonlinear one, we consider the self-exciting threshold autoregressive (SETAR) nonlinear modeling [41]. SETAR was chosen because it is a technique suitable for simulating complex and nonlinear patterns, and therewith, it is arguably closer to real-world scenarios (see [42]). In our study, to generate the synthetic series, we first fit the DPG's model to the `UNRATE` (unemployment rate data) quarterly series from the `Data_USEconModel` dataset available in the MATLAB software [43]. For this, we use the `auto.arima` function from the `forecast` R package [44] and the `selectSETAR` function of the `tsDyn` R package [45] for the linear and nonlinear DGPs, respectively. We then use these fit models to simulate combinations of 10, 101, and 500 time series with 60 and 222 observations using the `simulate` and `setar.sim` functions of the same `forecast` and `tsDyn` R packages mentioned previously. In all scenarios, the first half of the amount of all time series were set as the control units and the last half as the treated units.

For all scenarios, we set a prediction range of 12 and train our model using as a conditioning range the remaining dataset. Therefore, the intervention takes place at $T_0 = 49$ or $T_0 = 211$. The homogeneous intervention is emulated by adding, uniformly to the whole treated group, a constant corresponding to one standard deviation (sd) from the treated units before the intervention to all values of the treated units above the 0.9 quantiles, for all time steps after T_0. The idea is to simulate a scenario in which the intervention affects only a few quantiles of the treated units' distribution rather than its median or mean, as shown in Fig. 2. For the heterogeneous scenarios, the only difference is that we add a distinct fraction of the sd ($r \times sd$) for each treated series, where r is a different random value $r \in [0.7, 1.5]$ for each treated unit.

B. Empirical Study

As a real-world empirical illustration, we evaluate the impact of the increase in the number of ALI on EMSs demand attendances related to alcohol intoxication reported in one Australian state. We follow our earlier setup in [25], now focusing on the contribution of aggregating the quantile

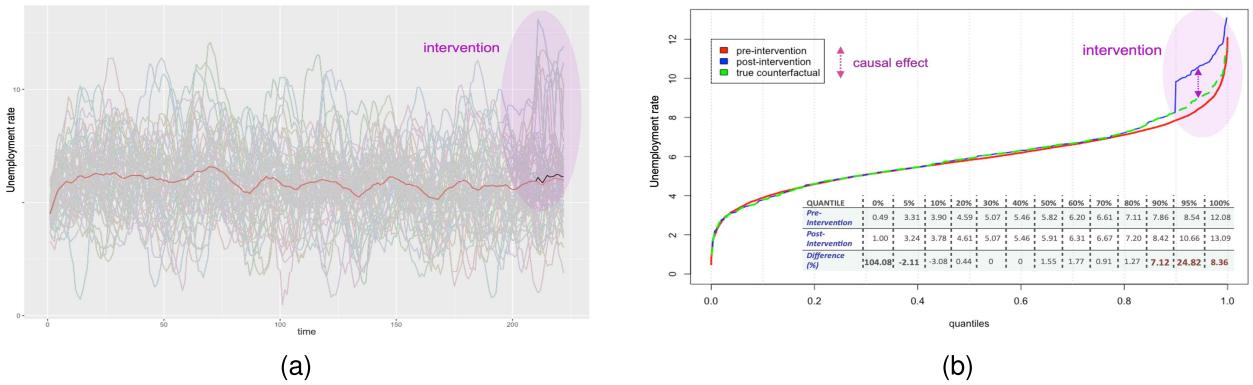

Fig. 2. (a) Simulated time series for the treated units: the gray lines are the simulated series, the red line the average of all time series, and the black line the average for the treated units after the intervention. (b) Quantile Distribution for the simulated treated units: preintervention and postintervention (observed values).

probabilistic forecasting with the proposed DeepProbCP. As the data distribution of EMS demand is strongly skewed, the new probabilistic approach should add additional and more accurate inferences about the impact of the ALI growth on EMS demand.

The National Ambulance Surveillance System (NASS), an ongoing public health surveillance system, is operated by Turning Point, an Australian addiction treatment and research center. The NASS [46] records surveillance data on alcohol and other drugs, self-harm, and mental health-related ambulance attendances across five of the six Australian states and two territories. The monthly EMS demand data relevant to the alcohol intoxication category (the "Alcohol Intox" variable from the NASS dataset) is available for all jurisdictions, but this report utilizes data from 79 local government areas (LGAs) in a single state. Data were included for the time period January 2015–May 2019 (53 months). The observations between June 2018 and May 2019 (12 months) were considered a test set because they correspond to the intervention period. There was an observed abnormal drift in the ALI time series for some local jurisdictions during July 2018–November 2018. Therefore, the preintervention time period was considered to be January 2015–May 2018 (41 months).

C. Benchmark Models and Performance Measuring

We benchmark the DeepProbCP against two well-established local structural causal models previously mentioned (ArCo and CausalImpact) and the other two global probabilistic forecasters (DeepAR and MQ-RNN). In addition, as part of the ablation study in order to evaluate the accuracy gains by increasing the framework complexity, we also compare the performance of DeepProbCP to a local multiple quantile estimator (MQ-RNN-local), a canonical standard feedforward NN (FFNN), a canonical global RNN, and our framework without the probabilistic layer (DeepCPNet). Except for the two first models, all others were implemented in the open-source GluonTS [47] software.

1) *ArCo:* A constrained LASSO regression-based model introduced by Carvalho *et* . [7] and implemented using the `ArCo` R package [48].
2) *CausalImpact:* A Bayesian structural time series method based on diffusion-regression state-space models developed by Brodersen *et al.* [9] and implemented via the `CausalImpact` R package [49].
3) *DeepAR:* A global Seq2Seq parametric probabilistic forecasting model using RNNs, proposed by Salinas *et al.* [17], which is trained with maximum likelihood estimation to output the parameters of distribution and, by applying sequential sampling, provides the probabilistic forecasts. This method requires the prespecification of the distribution that fits the data.
4) *MQ-RNN:* A global Seq2Seq nonparametric probabilistic forecaster model proposed by Wen *et* . [28], with an RNN as an encoder and a multiquantile MLP as a decoder, which jointly learns the quantile estimates over multiple quantiles without any explicit distribution assumption.
5) *MQ-RNN-Local:* A local version of the previous model where we pass the `Localizer` class to the global `MQRNNEstimator`.
6) *FFNN-Global:* A simple multilayer perceptron (MLP) model predicting the next target time-steps given the previous ones.
7) *RNN-Global:* A canonical RNN with LSTM as cell-type, however, without the stacked architecture.

By the ablation study, we intend to analyze, by the experimental results, the gains from the *local* to *global* modeling, from the shallow global NN architectures to deeper and more complex ones (FFNN to RNN), the use of the stacked architecture with peep-hole connections (RNN to DeepProbCP), the addition of the probabilistic layer (DeepCPNet to DeepProbCP), and the gains of migrating from a parametric probabilistic forecasting approach (DeepAR) to a nonparametric probabilistic approach (DeepProbCP and MQ-RNN). Finally, the difference between DeepProbCP and MQ-RNN relies on the use of Seq2Seq structures by MQ-RNN since, apart from that, both models are nonparametric probabilistic GFMs with RNNs.

To evaluate the point estimate prediction accuracy of the models, we report two scale-independent point error metrics that are commonly used in time series forecasting research. They are the symmetric Mean Absolute Percentage

Error (sMAPE) and the mean absolute scaled error (MASE). On the other hand, to quantify the quantile prediction accuracy of the algorithm, we compute the CRPS. The CRPS measures the compatibility of a forecast cumulative distribution function (represented by its quantile function $\widehat{\text{CDF}}^{-1}$) with a target observation z. It has an intuitive definition as the pinball loss [L_τ in (8)] integrated over all quantile levels $\tau \in [0, 1]$. This means that it averages the QLs over all quantiles, rather than optimizing a single QL, returning the accuracy of all the quantile estimations at once. The CRPS is extensively used within the forecasting literature as a probabilistic error metric to evaluate how probabilistic models get ground truth.

In the case of the causal/TE measurement accuracy, we employ only the sMAPE. All these error metrics are defined as follows:

$$\text{sMAPE} = \frac{2}{h}\sum_{t=n+1}^{n+h}\frac{|\hat{Y}_t - Y_t|}{|Y_t| + |\hat{Y}_t|} \tag{11a}$$

$$\text{MASE} = \frac{1}{h}\frac{\sum_{t=n+1}^{n+h}|\hat{Y}_t - Y_t|}{\frac{1}{n-S}\sum_{t=S+1}^{n}|Y_t - Y_{t-S}|} \tag{11b}$$

$$\text{CRPS} = \int_0^1 2L_\tau(\widehat{\text{CDF}}^{-1}(\tau), z)\,\mathrm{d}\tau. \tag{11c}$$

D. Hypothesis and Placebo Testing

We conduct hypothesis testing following the definitions and procedures presented in [8]. In that work, those authors introduce new inference procedures applicable to many different SCMs for policy evaluation when predicting counterfactual mean outcomes in the absence of policy intervention. Here, the main hypothesis of interest is

$$H_0 : \delta = 0 \tag{12}$$

where δ is the TE, as defined in (3). Alternatively, if we are interested in the ATE, (12) can be written as $H_0 : \Delta_T = 0$, with Δ_T as defined in (4).

For the treated units, we are interested in the rejection of H_0 to provide significant evidence about the existence of positive policy impacts. On the other hand, for the control units, we wait for the acceptance of H_0 to confirm the Assumption 2 about the null intervention effect over the control units—the placebo test. In other words, the placebo test is: when comparing the prediction errors of the estimator for treated units (TE) against the prediction errors for control units (postintervention forecast trajectories for the control units), the distribution of prediction errors for the treated units must be substantially larger (TE different from zero) relative to the distribution of prediction errors for the control units in the donor pool (null effect). The definitions of the test statistic and the p-value are given as follows:

$$S(\hat{u}) = \left(T_*^{-1/2}\sum_{t=T_0+1}^{T}|\hat{u}_t|\right) \tag{13}$$

$$\hat{p} = 1 - \hat{F}\big(S(\hat{u})\big), \quad \text{with } \hat{F}(x) = \frac{1}{\Pi}\sum_{\pi\in\Pi}\mathbf{1}\big\{S\big(\hat{u}_\pi\big) < x\big\} \tag{14}$$

where $\hat{u}_t$ is the error, the difference between the observed and predicted values, that is, the effect δ_t (for more details see [8]). We use the implementation of these tests in the R code provided by Chernozhukov *et al.* [8] in their online supplemental materials.

V. Results

A. Synthetic Data

Table I(a) shows the forecasting error metric results aggregated for all time series (treated and control groups). For the treated units, the errors refer to the comparison with the real counterfactual, which is recovered from the simulation. The table shows the best performance of DeepProbCP, considering the point estimate prediction accuracy (sMAPE and MASE), when forecasting the postintervention trajectories in almost all scenarios. We can see that the structural causal models, the local model, and the simpler NN structures (FFNN and RNN) always underperformed compared with the more complex probabilistic models (DeepProbCP, DeepAR, and MQ-RNN). DeepProbCP was dominant under the linear scenario, and under the nonlinear scenario, it was only not consistently better in a scenario with more data points (111 000), where the MQ-RNN performs better, and in one of the intermediate scenarios (with 6060 data points), with DeepAR presenting the best figures.

The accuracy results (CRPS) for the probabilistic quantile forecasting follow the same pattern, demonstrating the good ability to recover the real distribution. This result is corroborated by Fig. 3, where we observe the forecast quantile distributions following the true distribution closely, an indication that the model can capture the complex nature of the real distributions.

As previously shown in Fig. 2(b), the intervention took place only above the 0.9 quantiles, where almost no effect is seen over the median or mean of the treated units' distribution (approximately 2%) while the intervention over the right tail data of the distribution figures about 25% of increase. Therefore, here lies the usefulness of applying quantile probabilistic forecasting to predict the full distribution of the treated units. The results of building the quantile distribution using spline interpolation to better visualize the effects of this intervention are presented in Fig. 3 and Table I(b).

With this new approach, the estimated TE now is clearly visible in the last quantiles of the distribution and practically absent in the median. The size of the total TE (δ) in the quantile-probabilistic-spline forecasting is calculated by integrating the difference between the predicted observed (blue) and counterfactual (green or yellow) quantile distribution curves in Fig. 3. Computing these areas for each of the 12 forecast horizons of treated units, we have the residuals/causal effects, as reported in Table I(b) as "TTE." The table numerically demonstrated the TTE subestimation produced by the conventional nonprobabilistic forecasting methods (DeepCPNet, Causal Impact, and ArCo), with these models presenting the worst sMAPE for the TTE estimation. The big difference between the performances of DeepCPNet and DeepProbCP (which differs by the probabilistic feature) shows the gain of adding a probabilistic component to obtain more realistic causal impact predictions. DeepProbCP and

TABLE I

(a) ERROR MATRIC RESULTS FOR THE POINT AND PROBABILISTIC FORECASTING APPROACHES. (b) CASUAL EFFECT ESTIMATION AND THEIR RESPECTIVE ERROR METRIC RESULTS FOR THE POINT AND PROBABILISTIC FORECASTING APPROACHES

(a)

				Point Forecasting Error Metrics																Probabilistic Forecasting Error Metric							
				sMAPE								MASE								CPRS							
Ablation by the feature of the time-series set				Ablation by increasing the complexity of NN			Our Model	Probabilistic Models (Benchmark)		Structural Causal Models (Benchmark)		Ablation by increasing the complexity of NN			Our Model	Probabilistic Models (Benchmark)		Structural Causal Models (Benchmark)		Ablation by increasing the complexity of NN			Our Model	Probabilistic Models (Benchmark)		Structural Causal Models (Benchmark)	
DGP	No. Series	No. Dt Points	Length	MQ-RNN Local	FFNN	RNN	DeepProbCP	DeepAR	MQ-RNN	Causal Impact	ArCo	MQ-RNN Local	FFNN	RNN	DeepProbCP	DeepAR	MQ-RNN	Causal Impact	ArCo	MQ-RNN Local	FFNN	RNN	DeepProbCP	DeepAR	MQ-RNN	Causal Impact	ArCo
LINEAR ARIMA	Few: 10	60	Short: 60	0.2805	0.1702	0.1633	**0.1435**	0.1669	0.1915	0.2113	0.3078	1.6694	1.0793	1.0084	**0.8753**	0.9704	1.1092	1.153267	1.9687	0.1700	0.1089	0.1095	**0.1031**	0.1106	0.1307	NA	NA
		2220	Long: 222	0.2485	0.1746	0.1626	**0.1461**	0.1494	0.1759	0.1663	0.2877	1.4399	1.1099	0.9811	**0.8818**	0.9187	1.0636	1.018383	2.1287	0.1555	0.1187	0.1102	**0.0996**	0.1013	0.1201	NA	NA
	Interm: 101	6060	Short: 60	0.3006	0.2318	0.2052	**0.1713**	0.1895	0.1949	0.2461	0.2771	1.8933	1.5736	1.2686	**1.0460**	1.2084	1.2017	1.540229	1.8612	0.1986	0.1513	0.1311	**0.1043**	0.1210	0.1198	NA	NA
		22422	Long: 222	0.3953	0.1911	0.2033	**0.1711**	0.1907	0.2039	0.2865	0.3671	2.4279	1.2088	1.2484	**1.0464**	1.2070	1.3034	1.706337	2.2902	0.2486	0.1192	0.1290	**0.1036**	0.1244	0.1290	NA	NA
	Many: 500	30000	Short: 60	0.3347	0.1999	0.2049	**0.1727**	0.2215	0.1987	0.2107	0.2852	2.1310	1.2504	1.2959	**1.0692**	1.4826	1.2916	1.319959	1.9338	0.2077	0.1205	0.1281	**0.1002**	0.1435	0.1210	NA	NA
		111000	Long: 222	0.3294	0.1965	0.2065	**0.1714**	0.1964	0.1941	0.2440	0.2894	2.0068	1.2195	1.2655	**1.0327**	1.2174	1.2283	1.461038	1.9687	0.2014	0.1198	0.1302	**0.0974**	0.1220	0.1158	NA	NA
NON- LINEAR SETAR	Few: 10	60	Short: 60	0.3642	0.3265	0.3231	**0.3006**	0.3248	0.3355	0.3523	0.3667	1.8520	1.7976	1.7725	**1.5827**	1.7840	1.8186	1.8341	1.9221	0.2130	0.2102	0.2311	**0.1980**	0.2318	0.2275	NA	NA
		2220	Long: 222	0.3073	0.3047	0.3044	**0.2923**	0.3111	0.3332	0.3350	0.3085	1.4563	1.5136	1.4813	**1.4291**	1.5310	1.5872	1.6403	1.4499	0.2101	0.1954	0.2140	**0.2205**	0.2209	0.2472	NA	NA
	Interm: 101	6060	Short: 60	0.3805	0.2610	0.2791	0.2274	**0.2162**	0.2280	0.2781	0.2845	1.9052	1.4823	1.4790	1.2483	**1.2300**	1.2645	1.4957	1.5641	0.2440	0.1729	0.1957	0.1419	**0.1358**	0.1437	NA	NA
		22422	Long: 222	0.4068	0.2622	0.2798	**0.1989**	0.2152	0.2438	0.3401	0.2662	1.9102	1.3527	1.4811	1.1253	**1.0934**	1.3268	1.8909	1.5200	0.2411	0.1754	0.1966	**0.1332**	0.1395	0.1689	NA	NA
	Many: 500	30000	Short: 60	0.3782	0.2699	0.2825	**0.2317**	0.2456	0.2331	0.2521	0.2571	1.9372	1.5213	1.5383	**1.2844**	1.3513	1.2907	1.4349	1.5027	0.2383	0.1773	0.1993	**0.1546**	0.1620	0.1554	NA	NA
		111000	Long: 222	0.3632	0.2647	0.2828	0.2304	0.2237	**0.2211**	0.3061	0.2528	1.7981	1.4322	1.4672	1.1843	1.1863	**1.1468**	1.5912	1.3249	0.2364	0.1761	0.1988	0.1509	0.1510	**0.1461**	NA	NA

(b)

						Total Treatment Effect - TTE (treated units)									sMAPE for TTE estimation (calculated for each of 12 effects estimated and summarised by mean)									*p-values** (treated / control) alpha = 0.05 (H0 : TE = 0)		
						True total TE (q>90%)	Ablation by increasing the complexity of NN				Our Model	Probabilistic Models (Benchmark)		Structural Causal Models (Benchmark)	Ablation by increasing the complexity of NN				Our Model	Probabilistic Models (Benchmark)		Structural Causal Models (Benchmark)	Our Model	Probabilistic Models (Benchmark)	
MODEL:							MQ-RNN (local)	FFNN (global)	RNN (global)	DeepCPNet q=50% (global)	DeepProbCP (global)	DeepAR (global)	MQ-RNN (global)	Causal Impact (local)	ArCo (local)	MQ-RNN (local)	FFNN (global)	RNN (global)	DeepCPNet q=50% (global)	DeepProbCP (global)	DeepAR (global)	MQ-RNN (global)	Causal Impact (local)	ArCo (local)	DeepProbCP (global)	DeepAR (global)	MQ-RNN (global)
Scenario:	DGP	Kind of Treatment	Number of Time-series	Number of data points	Length of Time-series																					
	Linear - ARIMA	Homogenous Treatment Effect	Few: 10	600	Short: 60	66.57	-129.63	-32.65	31.53	4.30	116.31	137.30	177.68	9.25	8.76	1.3122	0.7344	**0.4591**	1.7574	0.6115	0.6666	0.8509	1.5120	1.5348	< 0.05 / 0.34	0.05 / 0.36	< 0.05 / 0.18
				2220	Long: 222	60.54	-9.35	-107.00	23.65	4.73	110.70	83.08	149.45	-0.25	-1.47	0.7500	1.0001	**0.5175**	1.7103	0.6123	0.6659	0.8296	2.0000	2.0000	< 0.05 / 0.30	< 0.05 / 0.32	< 0.05 / 015
			Interm: 101	6060	Short: 60	107.58	108.62	-330.19	67.24	7.64	135.82	187.98	144.62	3.58	4.50	0.9886	1.5912	0.8026	1.7347	**0.5661**	0.6779	0.5689	1.8711	1.8392	< 0.05 / 0.38	< 0.05 / 0.31	< 0.05 / 0.37
				22422	Long: 222	95.07	173.30	-106.69	25.99	7.33	123.46	185.71	138.32	5.00	5.84	0.8971	1.1728	0.8287	1.7136	**0.6057**	0.7649	0.6084	1.8002	1.7684	< 0.05 / 0.33	< 0.05 / 0.23	< 0.05 / 0.63
			Many: 500	30000	Short: 60	110.26	44.02	-333.91	49.58	3.08	161.93	193.67	163.74	3.83	2.91	0.6373	0.7901	0.7377	1.8912	**0.4354**	0.7235	0.4490	1.8657	1.8972	< 0.01 / 0.56	< 0.05 / 0.28	< 0.01 / 0.81
				111000	Long: 222	101.36	26.13	-301.67	55.29	3.09	153.21	167.30	135.10	2.44	2.03	0.6234	0.9359	0.8383	1.8818	0.5464	0.7070	**0.4525**	1.9060	1.9216	< 0.05 / 0.40	< 0.05 / 0.28	< 0.01 / 0.79
		Heterogenous Treatment Effect	Few: 10	600	Short: 60	78.50	-117.69	-20.72	43.46	4.30	128.24	149.24	189.61	9.85	9.36	1.1820	0.6838	**0.4140**	1.7924	0.5608	0.6101	0.7862	1.5540	1.5737	< 0.05 / 0.33	< 0.05 / 0.23	< 0.05 / 0.18
				2220	Long: 222	71.39	1.50	-96.14	34.51	4.73	121.56	93.93	160.31	0.29	-0.92	0.6940	0.9146	**0.4524**	1.7516	0.5622	0.6196	0.7687	1.9835	2.0000	< 0.01 / 0.78	< 0.05 / 0.47	< 0.05 / 0.22
			Interm: 101	6060	Short: 60	113.88	114.91	-323.90	73.54	7.64	142.12	194.27	150.91	3.77	4.69	0.9772	1.5545	0.8006	1.7485	**0.5608**	0.6695	0.5631	1.8718	1.8417	< 0.01 / 0.82	< 0.01 / 0.51	< 0.01 / 0.55
				22422	Long: 222	100.37	178.60	-101.38	31.29	7.33	138.76	191.01	143.62	5.16	6.01	0.8877	1.1542	0.8223	1.7277	**0.6104**	0.7486	0.6112	1.8043	1.7741	< 0.01 / 0.58	< 0.01 / 0.49	< 0.01 / 0.60
			Many: 500	30000	Short: 60	126.73	60.49	-317.43	66.05	3.08	178.41	210.14	180.21	4.19	2.03	0.6288	0.7539	0.7212	1.9050	**0.4012**	0.6553	0.4054	1.8721	1.9370	< 0.01 / 0.81	< 0.01 / 0.62	< 0.01 / 0.81
				111000	Long: 222	116.21	40.97	-286.83	70.13	3.09	168.05	182.14	149.94	2.77	2.35	0.6156	0.8678	0.8176	1.8965	0.4870	0.6358	**0.3945**	1.9070	1.9205	< 0.01 / 0.84	< 0.01 / 0.69	< 0.01 / 0.84
	Nonlinear - SETAR	Homogenous Treatment Effect	Few: 10	600	Short: 60	40.55	-80.50	-226.99	-92.44	10.74	-0.95	-12.70	10.62	-5.93	-5.82	1.1295	1.9530	1.3546	1.1624	1.2704	1.0808	**0.5098**	2.0000	2.0000	0.08 / 0.11	0.05 / 0.13	< 0.05 / 0.19
				2220	Long: 222	38.67	-77.20	-195.62	-55.50	7.18	14.04	-42.66	40.28	-8.82	-8.43	1.2056	1.9714	1.0211	1.3740	0.7919	1.1102	**0.6009**	2.0000	2.0000	< 0.05 / 0.26	0.06 / 0.17	< 0.05 / 0.39
			Interm: 101	6060	Short: 60	73.22	-5.40	-217.28	-0.40	6.17	102.06	58.81	42.41	6.10	5.05	0.9468	1.7892	0.7109	1.6891	**0.4904**	0.5777	0.6325	1.6926	1.7421	< 0.05 / 0.38	< 0.05 / 0.30	< 0.05 / 0.21
				22422	Long: 222	63.21	36.20	-114.89	19.09	7.06	101.86	117.87	113.75	6.15	5.38	0.7628	1.5350	0.9880	1.5980	**0.5229**	0.6861	0.6273	1.6453	1.6864	< 0.05 / 0.33	< 0.05 / 0.25	< 0.05 / 0.30
			Many: 500	30000	Short: 60	63.22	-30.57	-273.05	-9.00	3.67	113.97	125.33	87.24	0.63	0.64	0.8646	1.7146	0.7268	1.7805	0.6474	0.6902	**0.3955**	1.9603	1.9599	< 0.01 / 0.48	< 0.01 / 0.50	< 0.01 / 0.81
				111000	Long: 222	64.44	-33.65	-263.76	-27.20	3.74	114.85	-1.50	82.16	1.98	1.51	1.0595	1.8439	0.6949	1.7804	0.6072	0.6123	**0.3706**	1.8807	1.9082	< 0.01 / 0.65	< 0.01 / 0.59	< 0.01 / 0.87
		Heterogenous Treatment Effect	Few: 10	600	Short: 60	52.57	-68.47	-214.97	-80.42	10.74	11.07	-0.68	22.64	-5.32	-5.21	0.9555	1.8636	1.1631	1.3215	0.9989	0.8082	**0.4139**	2.0000	2.0000	0.07 / 0.13	0.06 / 0.16	< 0.05 / 0.58
				2220	Long: 222	50.14	-65.74	-184.16	-44.04	7.18	25.50	-31.20	7.57	-8.24	-7.85	1.0226	1.9210	0.8570	1.4992	0.8675	0.9087	**0.5145**	2.0000	2.0000	< 0.05 / 0.18	< 0.05 / 0.27	< 0.05 / 0.63
			Interm: 101	6060	Short: 60	78.49	-0.13	-212.01	4.87	6.17	107.33	64.08	47.68	6.27	5.92	0.9190	1.7839	0.6984	1.7084	**0.4659**	0.5617	0.6117	1.7042	1.7196	< 0.05 / 0.82	< 0.05 / 0.51	< 0.05 / 0.64
				22422	Long: 222	73.48	2.03	-149.05	-15.07	7.06	67.70	83.71	79.59	6.99	6.57	0.7640	1.5979	0.7297	1.6492	**0.1981**	0.5670	0.2376	1.6525	1.6716	< 0.01 / 0.90	< 0.01 / 0.52	< 0.01 / 0.83
			Many: 500	30000	Short: 60	73.90	-19.88	-262.36	1.68	3.67	124.66	116.02	97.93	0.81	0.82	0.8229	1.6746	0.7222	1.8107	0.5973	0.5888	**0.3646**	1.9566	1.9563	< 0.01 / 0.73	< 0.01 / 0.52	< 0.01 / 0.90
				111000	Long: 222	75.35	-22.74	-252.84	-16.29	3.74	125.77	9.41	93.07	2.16	1.69	0.9856	1.8311	0.6925	1.8107	0.5663	0.6132	**0.3361**	1.8884	1.9121	< 0.01 / 0.74	< 0.01 / 0.65	< 0.01 / 0.89

* *p-value* reported only for the 3 best models

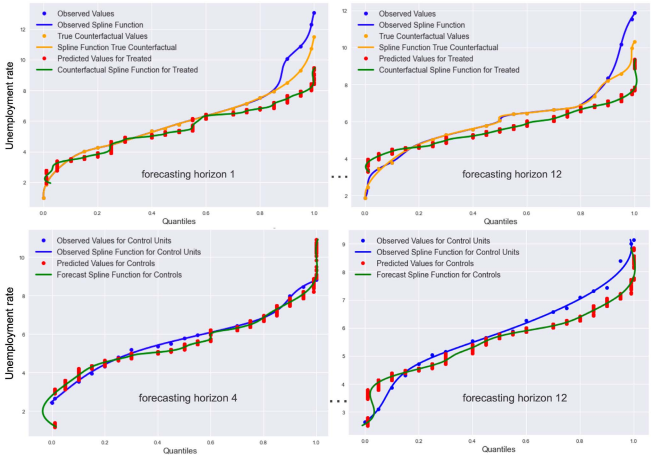

Fig. 3. Quantile distribution of simulated treated and control units for the forecasting horizons 1, 4, and 12 built by the spline interpolation of 21 quantile forecasts.

MQ-RNN produce better TTE estimations than DeepAR, demonstrating the better performance of nonparametric probabilistic models for TTE estimation under these nonuniform intervention circumstances. Once more, MQ-RNN performs consistently better in the scenarios with more data points and DeepProbCP better under the linear scenarios. The results confirm the improved accuracy and adequacy when employing the quantile-probabilistic approach in scenarios where the intervention does not have a uniform effect over the distribution.

Finally, to confirm the null effect of the intervention over the control units, we performed the hypothesis testing, as described in Section IV-D, but only for the three best models (DeepProbCP, DeepAR, and MR-RNN). The null hypothesis of a zero effect $H_0 : \delta_1 = \delta_2 = \cdots = \delta_{12}$ was not rejected for control units for all scenarios and rejected for treated units for almost all scenarios, confirming the positiveness of the placebo tests and the significance of the TE for the treated units, as reported in last columns of Table I(b).

B. Real-World Data

Previously, to estimate the effect of the increase in ALI over the number of ambulance calls related to alcohol intoxication, from Fig. 4, we depict the fact that the EMS dataset is strongly skewed and how the intervention was concentrated in the higher quantiles. We conjecture that this effect can be attributed to the different geographical spans of the LGAs, where larger ones account for a higher growth of ALI.

First, we assess the forecasting performance of the different algorithms by point estimating the counterfactuals only for the control units. In the absence of true underlying effects, in contrast to the simulation study, this process allows us to establish a baseline of prediction errors for assessing the TE later. These errors for control units need to be low and statistically significantly different from the error gaps of the treated units (the placebo test). On the other hand, the error gap of the treated units, the causal effect, must be statistically significantly different from zero. The testing follows the approach outlined in Section IV-D.

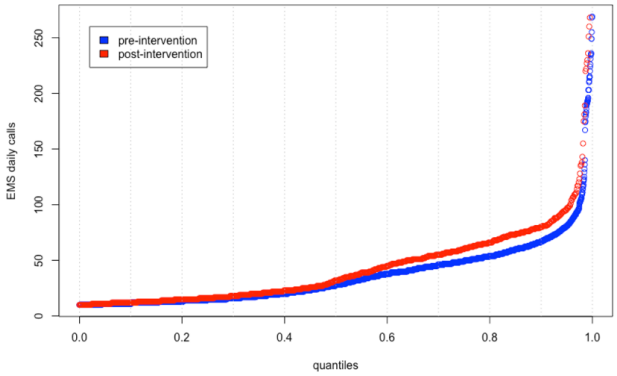

Fig. 4. NASS dataset: quantile distribution for the 69 treated units—observed data.

TABLE II
ERROR RESULTS ONLY FOR THE CONTROL UNIT SERIES OF THE NASS DATASET

Method (for Control Group)	sMAPE (point estimation metrics)	MASE (point estimation metrics)	CRPS (probabilistic metric)
MQRNN-local	0.7153	4.5646	0.3443
	(6.3819)	*(6.3819)*	*(0.1719)*
DeepAR-local	0.1777	1.3234	0.1367
	(0.0593)	*(0.3901)*	*(0.0522)*
FFNN	0.1938	1.7438	0.1801
	(0.0433)	*(0.1845)*	*(0.0487)*
RNN	0.1448	1.0316	0.1043
	(0.0140)	*(0.4950)*	*(0.0431)*
DeepProbCP	**0.1340**	0.9917	**0.1008**
	(0.0463)	*(0.2635)*	*(0.0367)*
DeepAR	0.13667	**0.9783**	0.1129
	(0.0395)	*(0.2253)*	*(0.0466)*
MQRNN	0.1492	1.1272	0.1015
	(0.0517)	*(0.2592)*	*(0.0462)*
CausalImpact	0.1466	0.7882	NA
	(0.0456)	*(0.6691)*	NA
ArCo	0.1423	0.7676	NA
	(0.0503)	*(0.6492)*	NA
Season Naive	0.1686	1.2363	0.1652
	(0.0533)	*(0.4126)*	*(0.0487)*

The point estimate prediction accuracies are reflected by the MAPE and MASE values reported in Table II (where the values inside the parenthesis refer to the sd). Accordingly to the simulated study, we can see how the more complex global models perform better than the local and less complex ones. DeepProbCP presents one of the best performances. This gives us the confidence to use it to estimate the causal effect for the treated group. Fig. 5 shows visually the counterfactual prediction and the median effect size (per LGA) throughout the LGAs that experienced the increase in ALI. We can observe the lower impact over the control group, which is confirmed by the p-values reported in Table III. However, the point estimation forecasting is only reflecting the median effect. By Fig. 4, we see that the intervention was concentrated on the upper quantiles, motivating the TE estimation by the probabilistic framework in order to obtain a more accurate measure of the impact size.

Second, we perform the probabilistic prediction of the counterfactual quantile distribution to deepen the estimation of this TE by quantiles and observe better the dynamics of the causal effect distribution (a sample of results is shown in Fig. 6). The probabilistic forecasting accuracy of the proposed

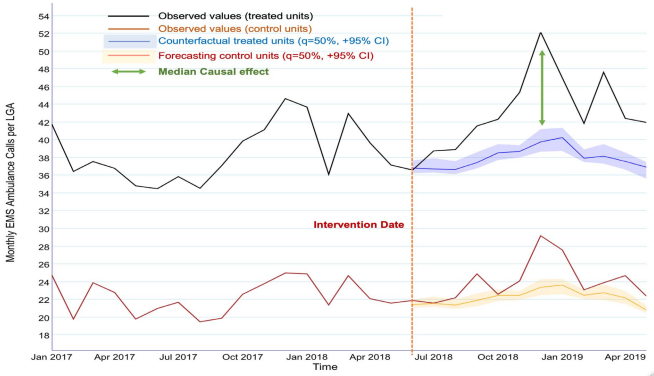

Fig. 5. NASS dataset: observed and forecast values aggregated by mean for the 69 treated units (upper curve) and the ten control units (lower curve). The shaded areas correspond to the quantile predictions. The difference between the observed trajectory for the treated units and its counterfactual outcome is considered the causal effect of the intervention.

TABLE III

ESTIMATION OF THE TE OF LIQUOR LICENSE (ALI) INCREASE FOR AFFECTED LGAS IN THE NASS DATASET

Probabilistic quantile forecasting: DeepProbCP		
quantiles	**Average Treatment Effect**	**% Growth of Ambulance Calls**
0.01	82.3013	3.14
0.05	108.5240	4.14
0.10	135.8839	5.19
0.20	196.3658	7.50
0.30	251.6996	9.61
0.40	304.2688	11.62
0.50	351.8395	13.43
0.60	394.8470	15.08
0.70	435.5513	16.63
0.80	478.7610	18.28
0.90	521.3871	19.91
0.95	523.6383	19.99
0.99	526.7686	20.11
Point estimate forecasting: DeepCPNet (tradicional approach)	351.8395	13.43
p-values (treated / controls)	0.012 / 0.278	

DeepProbCP can be demonstrated, first, by its distribution recovery ability. In Fig. 6, we observe that the forecast counterfactual quantile function follows the shape of the true distribution closely. This indicates that the model is able to capture the complex nature of the true distribution even without information about it (apart from the actual time series inputs). Second, we assess the performance of the quantile prediction accuracy by the CRPS metric reported in Table II, whose results show that the quantile forecasting of DeepProbCP is one of the best among the baseline models. Another evidence of its probabilistic accuracy can be seen by how the median forecast remains inside the predicted confidence intervals, as illustrated in Fig. 5.

Finally, Table III shows the difference between using the conventional point estimate approach and the probabilistic quantile forecasting to estimate the ATE of increasing the ALI over certain LGAs. As we can see, while the traditional method returns an average impact of about 13% of growth

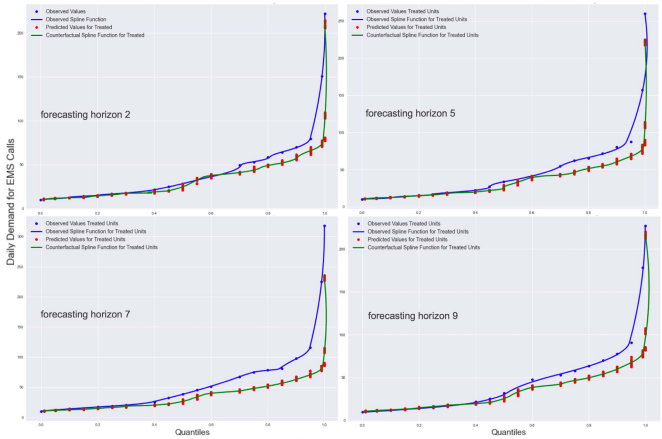

Fig. 6. NASS dataset: counterfactual quantile distributions predicted for the forecasting horizons 2, 5, 7, and 9.

in the number of ambulance calls, the probabilistic approach shows that this impact might be in extreme case almost double, namely, about 20% in the upper quantiles. Fig. 6 depicts a similar effect, where we observe a considerable impact on the maximum levels of ambulance calls for the forecast horizon 7.

The government could use this information to identify actual and precise alcohol/liquor license density costs. This approach gives the policymaker an improved approximation of the actual costs of issuing liquor licenses regarding community impact and harms, and the required funding of health services that respond to these harms. This precise effect inference could inform policy around alcohol license distribution and highlight the need for community involvement in decisions around issuing liquor licenses.

This more accurate TE estimation is vital for sensitive sectors, such as healthcare, where policies impact people's life. In this sense, the proposed framework might offer helpful guidance for forwarding planning on efficient allocation of resources for ambulance services for the worst case scenarios in areas where additional liquor licenses got issued. For the ambulance services management, it is also important to anticipate likely demand increments caused by growth in ALI to prepare their staff and avoid problems in the response time for attendance (overtime costs), which might ultimately cost lives.

There is a host of possibilities of analysis that also go beyond merely computing counterfactuals and measuring the effects. More than providing a comprehensive analysis of this particular case of potential costs caused in the health system by the increase of liquor licenses, our objective was to introduce a novel general technique and way to perform causal effect analysis under more complex and subtle or shaded scenarios of policy intervention.

VI. CONCLUSION

We have proposed a framework to model conditional quantile functions using splines to build counterfactual outcomes to be able to analyze the effects provoked by nonuniform interventions or where the data present non-Gaussian characteristics, such as skewness and heavy tails.

Recent research related to this subject, such as quantile-probabilistic forecasting and *global* NN forecasting applied to causal inference, have so far not been combined to study the uncertainty behind the causal impact analysis in more complex and nonparametric scenarios of intervention when several time series are involved.

We have presented a global RNN-based probabilistic forecasting modeling framework to fill this gap and enrich the decision-making process with powerful tools capable of not only delivering point predictions of counterfactuals but also predictions of their distributions. We have demonstrated the effectiveness of the approach on synthetic and real-world datasets, where the results indicate that we are able to outperform well-established structural causal models.

Some limitations of our approach that could be addressed in future work are that we currently do not promote the optimization of the spline parameters while building the estimated quantile distributions or performing the global RNN network engine simultaneously for all quantiles' forecasts. This could further boost the performance of the framework. Furthermore, our current experimental framework does not consider tasks where external regressors are available.

Our *global*-RNN-based engine for counterfactual prediction has demonstrated the effectiveness of learning a *global* model from nonlinear related and nonrelated time series, handling a variety of scales, and mapping complex patterns. With this causality-inspired DNN model combined with nonparametric probabilistic techniques, we improve the generalization and adaptability of the framework by leveraging the causality analysis to achieve reliable causal effect identifications in real-world scenarios.

ACKNOWLEDGMENT

The authors would like to thank the Turning Point researchers who code the National Ambulance Surveillance System (NASS) data, and ambulance services and paramedics who create and provide that data.

REFERENCES

[1] S. Athey and G. W. Imbens, "Identification and inference in nonlinear difference-in-differences models," *Econometrica*, vol. 74, no. 2, pp. 431–497, Mar. 2006.

[2] T. G. Conley and C. R. Taber, "Inference with 'difference in differences'? With a small number of policy changes," *The Rev. Econ. Statist.*, vol. 93, no. 1, pp. 113–125, 2011.

[3] N. Doudchenko and G. W. Imbens, "Balancing, regression, difference-in-differences and synthetic control methods: A synthesis," Nat. Bureau Economic Res., Cambridge, MA, USA, Tech. Rep. w22791, 2016.

[4] A. Abadie, A. Diamond, and J. Hainmueller, "Synthetic control methods for comparative case studies: Estimating the effect of California's tobacco control program," *J. Amer. Stat. Assoc.*, vol. 105, no. 490, pp. 493–505, 2010.

[5] S. Athey, M. Bayati, N. Doudchenko, G. Imbens, and K. Khosravi, "Matrix completion methods for causal panel data models," *J. Amer. Stat. Assoc.*, vol. 116, pp. 1–15, Oct. 2021.

[6] L. Gobillon and T. Magnac, "Regional policy evaluation: Interactive fixed effects and synthetic controls," *Rev. Econ. Statist.*, vol. 98, no. 3, pp. 535–551, Jul. 2016.

[7] C. Carvalho, R. Masini, and M. C. Medeiros, "ArCo: An artificial counterfactual approach for high-dimensional panel time-series data," *J. Econometrics*, vol. 207, no. 2, pp. 352–380, Dec. 2018.

[8] V. Chernozhukov, K. Wüthrich, and Y. Zhu, "An exact and robust conformal inference method for counterfactual and synthetic controls," *J. Amer. Stat. Assoc.*, vol. 116, pp. 1–16, Oct. 2021.

[9] K. H. Brodersen, F. Gallusser, J. Koehler, N. Remy, and S. L. Scott, "Inferring causal impact using Bayesian structural time-series models," *Ann. Appl. Statist.*, vol. 9, no. 1, pp. 247–274, Mar. 2015.

[10] D. B. Rubin, "Estimating causal effects of treatments in randomized and nonrandomized studies," *J. Educ. Psychol.*, vol. 66, no. 5, p. 688, Oct. 1974.

[11] G. E. Box, G. M. Jenkins, G. C. Reinsel, and G. M. Ljung, *Time Series Analysis: Forecasting and Control*. Hoboken, NJ, USA: Wiley, 2008.

[12] R. Hyndman, A. B. Koehler, J. K. Ord, and R. D. Snyder, *Forecasting With Exponential Smoothing: State Space Approach*. Berlin, Germany: Springer, 2008.

[13] S. Smyl, "A hybrid method of exponential smoothing and recurrent neural networks for time series forecasting," *Int. J. Forecasting*, vol. 36, pp. 75–85, Jan. 2020.

[14] K. Bandara, C. Bergmeir, and S. Smyl, "Forecasting across time series databases using recurrent neural networks on groups of similar series: A clustering approach," *Expert Syst. Appl.*, vol. 140, Feb. 2020, Art. no. 112896.

[15] T. Januschowski *et al.*, "Criteria for classifying forecasting methods," *Int. J. Forecasting*, vol. 36, no. 1, pp. 167–177, Jan. 2020, doi: 10.1016/j.ijforecast.2019.05.008.

[16] P. Montero-Manso and R. J. Hyndman, "Principles and algorithms for forecasting groups of time series: Locality and globality," *Int. J. Forecasting*, vol. 37, no. 4, pp. 1632–1653, Oct. 2021, doi: 10.1016/j.ijforecast.2021.03.004.

[17] D. Salinas, V. Flunkert, J. Gasthaus, and T. Januschowski, "DeepAR: Probabilistic forecasting with autoregressive recurrent networks," *Int. J. Forecasting*, vol. 36, no. 3, pp. 1181–1191, Jul. 2020, doi: 10.1016/j.ijforecast.2019.07.001.

[18] J. Gasthaus *et al.*, "Probabilistic forecasting with spline quantile function RNNs," in *Proc. 22nd Int. Conf. Artif. Intell. Statist. (AISTATS)*, vol. 89, 2020, pp. 1901–1910.

[19] M. H. Farrell, T. Liang, and S. Misra, "Deep neural networks for estimation and inference," 2018, *arXiv:1809.09953*.

[20] J. Hartford, G. Lewis, K. Leyton-Brown, and M. Taddy, "Deep IV: A flexible approach for counterfactual prediction," in *Proc. ICML*, 2017, pp. 1414–1423.

[21] U. Shalit, F. D. Johansson, and D. Sontag, "Estimating individual treatment effect: Generalization bounds and algorithms," in *Proc. ICML*, 2017, pp. 3076–3085.

[22] B. Lim, "Forecasting treatment responses over time using recurrent marginal structural networks," in *Proc. NIPS*, 2018, pp. 7483–7493.

[23] C. Shi, D. Blei, and V. Veitch, "Adapting neural networks for the estimation of treatment effects," in *Proc. NIPS*, 2019, pp. 2507–2517.

[24] J. Poulos and S. Zeng, "RNN-based counterfactual prediction, with an application to homestead policy and public schooling," 2017, *arXiv:1712.03553*.

[25] P. Grecov *et al.*, "Causal inference using global forecasting models for counterfactual prediction," in *Proc. Pacific-Asia Conf. Knowl. Discovery Data Mining*. Cham, Switzerland: Springer, 2021, pp. 282–294.

[26] Y. Zeng, Z. Hao, R. Cai, F. Xie, L. Huang, and S. Shimizu, "Nonlinear causal discovery for high-dimensional deterministic data," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, Sep. 3, 2021, doi: 10.1109/TNNLS.2021.3106111.

[27] Y. Wen *et al.*, "Performance evaluation of probabilistic methods based on bootstrap and quantile regression to quantify PV power point forecast uncertainty," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 4, pp. 1134–1144, Apr. 2020.

[28] R. Wen, K. Torkkola, B. Narayanaswamy, and D. Madeka, "A multi-horizon quantile recurrent forecaster," 2017, *arXiv:1711.11053*.

[29] P. W. Holland, "Statistics and causal inference," *J. Amer. Stat. Assoc.*, vol. 81, no. 396, pp. 945–960, 1986.

[30] S. Athey and G. W. Imbens, "The state of applied econometrics: Causality and policy evaluation," *J. Econ. Perspect.*, vol. 31, no. 2, pp. 3–32, May 2017.

[31] B. Ferman and C. Pinto, "Revisiting the synthetic control estimator," Munich Pers. RePEc Arch., Munich, Germany, Tech. Rep. Item ID 86495, 2016.

[32] A. Steinkraus, "Estimating treatment effects with artificial neural nets: A comparison to synthetic control method," *Econ. Bull.*, vol. 39, no. 4, pp. 2778–2791, 2019.

[33] K. Bandara, C. Bergmeir, S. Campbell, D. Scott, and D. Lubman, "Towards accurate predictions and causal 'what-if' analyses for planning and policy-making: A case study in emergency medical services demand," in *Proc. IJCNN*, Jul. 2020, pp. 1–10.

[34] H. Hewamalage, C. Bergmeir, and K. Bandara, "Recurrent neural networks for time series forecasting: Current status and future directions," *Int. J. Forecasting*, vol. 37, no. 1, pp. 388–427, 2021.
[35] S. M. Stigler, "The transition from point to distribution estimation," *Bull. Int. Stat. Inst.*, vol. 46, pp. 332–340, Feb. 1975.
[36] T. Gneiting, "Quantiles as optimal point forecasts," *Int. J. Forecasting*, vol. 27, no. 2, pp. 197–207, Apr./Jun. 2011, doi: 10.1016/j.ijforecast.2009.12.015.
[37] T. Gneiting and M. Katzfuss, "Probabilistic forecasting," *Annu. Rev. Statist. Appl.*, vol. 1, pp. 125–151, Jan. 2014.
[38] K. Bandara, C. Bergmeir, and H. Hewamalage, "LSTM-MSNet: Leveraging forecasts on sets of related time series with multiple seasonal patterns," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 4, pp. 1586–1599, Apr. 2021.
[39] C. Runge, "Über empirische funktionen und die interpolation zwischen äquidistanten ordinaten," *Zeitschrift Math. Phys.*, vol. 46, nos. 224–243, p. 20, 1901.
[40] (2021). *Univariatespline*. [Online]. Available: https://docs.scipy.org/doc/scipy/reference/generated/scipy.interpolate.UnivariateSpline.html
[41] E. Hanifi Firat, "SETAR (self-exciting threshold autoregressive) nonlinear currency modelling in EUR/USD, EUR/TRY and USD/TRY parities," *Math. Statist.*, vol. 5, no. 1, pp. 33–55, Jan. 2017.
[42] H. Hewamalage, C. Bergmeir, and K. Bandara, "Global models for time series forecasting: A simulation study," *Pattern Recognit.*, vol. 124, Apr. 2022, Art. no. 108441.
[43] I. The MathWorks. (1994). *Simulate VaR Model Conditional Responses*. MathWorks R2021b. [Online]. Available: https://au.mathworks.com/help/econ/simulate-var-model-conditional-responses.html
[44] R. J. Hyndman *et al.* (2020). *Package 'Forecast'*. [Online]. Available: https://cran.r-project.org/web/packages/forecast/forecast.pdf
[45] A. F. Di Narzo, J. L. Aznarte, M. Stigler, and H. Tsung-Wu. (Feb. 2020). *tsDyn: Nonlinear Time Series Models With Regime Switching. R Package Version 10-1.2*. [Online]. Available: https://cran.r-project.org/web/packages/tsDyn/index.html
[46] I. Dan Lubman *et al.*, "The national ambulance surveillance system," *PLoS ONE*, vol. 15, Jan. 2020, Art. no. e0228316.
[47] A. Alexandrov *et al.*, "GluonTS: Probabilistic time series models in Python," 2019, *arXiv:1906.05264*.
[48] Y. R. Fonseca, R. Masini, M. C. Medeiros, and G. F. R. Vasconcelos. (Nov. 2017). *Arco: Artificial Counterfactual Package. R Package Version 0.3-1*. [Online]. Available: https://CRAN.R-project.org/package=ArCo
[49] K. H. Brodersen and A. Hauser. (2014). *Causalimpact—An R Package for Causal Inference Using Bayesian Structural Time-Series Models. R Package Version 1.2.7*. [Online]. Available: https://cran.r-project.org/web/packages/CausalImpact/vignettes/CausalImpact.html

Priscila Grecov received the B.Sc. degree in economics from the University of Brasília, Brasília, Brazil, in 1996, the master's degree in data science from Monash University, Melbourne, VIC, Australia, in 2020, and the M.Sc. degree in mathematics from the Institute of Pure and Applied Mathematics, Rio de Janeiro Brazil, in 2005. She is currently pursuing the Ph.D. degree in computer science with the Faculty of Information Technology, Monash University.

Her research interests include causal inference, deep neural networks, and time series forecasting using machine learning methods.

Ankitha Nandipura Prasanna received the B.E. degree in computer science and engineering from Visvesvaraya Technological University, Belgaum, India, in 2014. She is currently pursuing the master's degree in artificial intelligence with the Faculty of Information Technology, Monash University, Melbourne, VIC, Australia.

Her research interests include deep neural networks and time series forecasting.

Klaus Ackermann received the B.Sc. and M.Sc. degrees in business informatics with a major in economics from the Vienna University of Technology, Vienna, Austria, and the Ph.D. degree in economics from Monash University, Melbourne, VIC, Australia, in 2010, 2012, and 2016, respectively.

He is currently a Lecturer with the Department of Econometrics and Business Statistics, Monash University. He is also a Founding Member of SoDa Labs, an empirical research laboratory associated with the Monash Business School. His research interests are in the areas of data science, machine learning, public policy, and applied econometrics.

Sam Campbell is currently a Data Science Lead of the National Addiction and Mental Health Surveillance Unit, Monash University, a Senior Data Scientist with Monash University, Melbourne, VIC, Australia, and a Google AI Impact Challenge Grantee.

Debbie Scott is currently an Associate Professor and a Public Health Researcher with Monash University, Melbourne, VIC, Australia, where she is also the Strategic Lead of the National Addiction and Mental Health Surveillance Unit, Turning Point.

Dan I. Lubman is currently the Executive Clinical Director of Turning Point, Australia's National Addiction Treatment, Training and Research Centre, the Director of the Monash Addiction Research Centre, and a Professor of addiction studies and services with Monash University, Melbourne, VIC, Australia. He is also a psychiatrist and an addiction medicine specialist.

Christoph Bergmeir received the M.Sc. degree in computer science from the University of Ulm, Ulm, Germany, and the Ph.D. degree in computer science from the University of Granada, Granada, Spain, in 2010 and 2013, respectively.

He is currently a Senior Lecturer in data science and artificial intelligence with the Department of Data Science and Artificial Intelligence, Monash University, Melbourne, VIC, Australia. He also works as a data scientist in a variety of projects with external partners in diverse sectors, such as sustainable energy and supply chain. He has led teams that have delivered systems for short-term power production forecasting of wind and solar farms, and energy price forecasting. He has over 3800 citations and an H-index of 24.

Dr. Bergmeir received more than AUD 2.7 million in external research funding. Four of his publications on time series forecasting over the last years have been Clarivate Web of Science Highly Cited Papers (top 1% of their research field).